ADAPTIVE STEP-SIZE SIGN LEAST MEAN SQUARES

Mansour A. Aldajani

Systems Engineering Department
King Fahd University of Petroleum and Minerals
Dhahran 31261, Saudi Arabia
dajani@ccse.kfupm.edu.sa

ABSTRACT

In this paper, we propose a novel structure for adaptive sign least mean squares (ASLMS). A powerful adaptation scheme is used to adapt the step-size of the sign function inside the recursion of the sign algorithm. It is shown how the algorithm can be implemented with no real multiplication. Simulation result show that the performance of the proposed algorithm can be made arbitrarily close to that of the original least means squares algorithm.

1. INTRODUCTION

Least mean squares (LMS) algorithm is widely used and well-established adaptive filtering algorithm. LMS is used to estimate parameters or weights from a measured process. The weights update of LMS is given by the recursion [1]

$$\begin{aligned} w_i &= w_{i-1} + \mu u_i^* \left[d(i) - u_i w_{i-1} \right], \quad i \geq 0 \qquad (1) \\ w_0 &= \text{initial weight vector} \end{aligned}$$

where

$$\begin{aligned} w_i &: M \times 1 \text{ weight vector at iteration } i \\ u_i &: 1 \times M \text{ regression vector} \\ d(i) &: \text{scalar desired signal} \\ \mu &: \text{fixed adaptation step} - \text{size} \end{aligned}$$

and the star (*) indicates the conjugate transpose. The desired signal $d(i)$ is assumed to be drawn from a process with desired optimum weights w_d with additive process noise $v(i)$, i.e.,

$$d(i) = u_i w_i + v(i). \qquad (2)$$

The a priory estimation error $e_a(i)$ is defined as

$$e_a(i) \triangleq d(i) - u_i w_{i-1} \qquad (3)$$

while the a posteriori estimation error $e_p(i)$ is given by

$$e_p(i) \triangleq d(i) - u_i w_i. \qquad (4)$$

The mean-square error (MSE) is defined here as

$$MSE \triangleq E|e_p(i)|^2. \qquad (5)$$

Other versions of LMS are also widely available (such as Sign-LMS (SLMS)[1], Sign-Sign LMS (SSLMS), Normalized LMS (NLMS), and Recursive Least Squares (RLS)) that offer compromise between performance and complexity.

Sign LMS is similar to LMS except that the adaptation rule involves only the sign of the a-priory error, namely,

$$\begin{aligned} w_i &= w_{i-1} + \mu u_i^* \operatorname{sign}\left[d(i) - u_i w_{i-1} \right], \quad i \geq 0 \qquad (6) \\ w_0 &= \text{initial weight vector} \end{aligned}$$

This equation can be represented in a block diagram as shown in Fig. 1. The error between the desired output $d(i)$ and the actual output $\widehat{d}(i) \equiv u_i w_{i-1}$ is measured. The error sign $s(i)$ is then computed and multiplied by both the fixed step-size μ and the conjugate vector u_i^*. The result is then accumulated to produce the updated weight vector. The thick lines indicate vector operations while the thin lines indicate scalar operations.

SLMS provides simpler implementation than LMS mainly because it involves sign multiplication. This is especially important for high speed applications where hardware implementation is necessary [2]. Rigorous analysis of the sign LMS is covered in [3] and [4].

Looking closely at Fig. 1, we notice that the structure of SLMS algorithm is similar to that of a *linear delta modulator* [2] shown in Fig. 2. Similar to delta modulators, one problem with SLMS is that the signum function introduces large *quantization* error, which is usually manifested in large MSE. This fact becomes a problem when the optimum desired filter tap weights are small in amplitude. In this case, the

This work was supported by King Fahd University of Petroleum and Minerals.

[1]Sign-LMS is also known as the sign algorithm (SA).

[2]This is more evident when you consider the scalar case with $u(i)$=1.

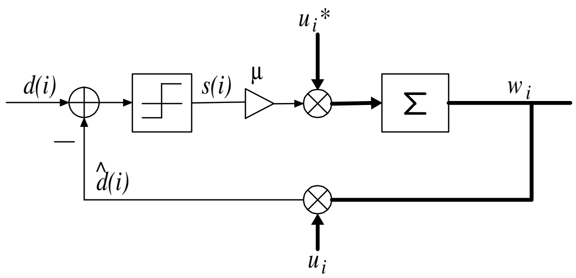

Fig. 1. *Block-diagram representation of the SLMS algorithm.*

noise created by the signum function becomes evidently noticeable compared to the amplitude of the filter tap weights, limiting the dynamic range of the adapter. One way to reduce these errors is to use small adaptation step-size, which usually slows down the convergence of the algorithm [5].

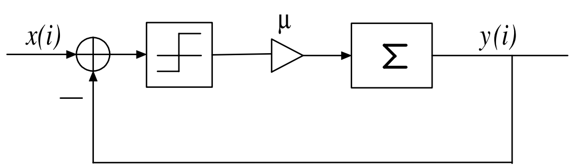

Fig. 2. *Linear Delta Modulator.*

2. STEP-SIZE ADAPTATION

In a previous study, a new step-size adaptation scheme was proposed [6]. The scheme is shown in Fig. 3. The adaptation loop (the dashed box) is simply trying to track the absolute value information of the input to the single-bit quantizer Q_1. In other words, the signal $s(i)$ carries the sign information of the input $e_a(i)$, while the absolute value information is *tracked* by the signal $f(i)$. As a result, the multiplication

$$v(i) = s(i)f(i)$$

should approximate the input $e_a(i)$. This adaptation scheme was impeded in two applications, namely, delta modulation and sigma delta modulation [7, 8]. The adaptation scheme improved the SNR, convergence speed, and dynamic range of these modulators.

In this work, we apply the step-size adaptation scheme of Fig. 3 to the sign algorithm. More specifically, the sign block of the sign algorithm shown in Fig. 1 will be replaced by this adaptation scheme. The purpose of this alteration is to reduce the impact of the sign block on the performance, getting as close as possible to the behavior of the original LMS algorithm.

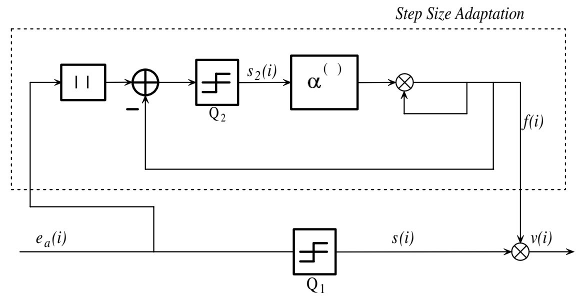

Fig. 3. *Step-size adaptation scheme.*

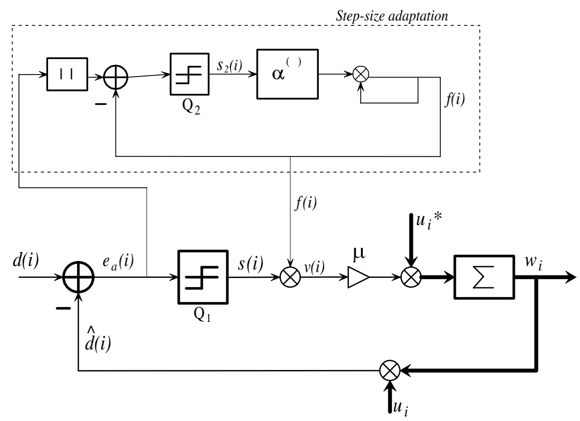

Fig. 4. *The proposed adaptive step-size sign algorithm.*

3. STRUCTURE OF THE ADAPTIVE SIGN ALGORITHM

The structure for the proposed adaptive sign algorithm is shown in Fig. 4. The structure is similar to that of the sign algorithm of Fig. 1 except that the step-size of the sign function (or single bit quantizer) is now adapted. The adaptation scheme discussed in Sec. 2 and presented in Fig. 3 is used for this purpose. The equations governing the dynamics of the step-size adaptation are

$$s(i) = \text{sign}\left[e_a(i)\right] \quad (7)$$

$$s_2(i) = \frac{\Delta}{2}\text{sign}\left[|e_a(i)| - f(i-1)\right] \quad (8)$$

$$f(i) = \alpha^{s_2(i)} f(i-1), \quad f(0) = \text{initial guess} \quad (9)$$

$$v(i) = s(i)f(i). \quad (10)$$

The constant Δ is the quantization step-size of Q_2. It can be verified that equation (9) can also be written as

$$f(i) = \alpha^{p(i)} \quad (11)$$

with

$$p(i) = p(i-1) + s_2(i), \quad p(0) = \text{initial guess}. \quad (12)$$

We will show in section 5 that by proper choice of the exponent constant α inside the adaptation loop and the fixed step-size μ, one can achieve a performance close to that of the original LMS algorithm. In the next section, we will show how this can be achieved without compromising the advantage of the sign algorithm, namely, the multiplication-free recursion.

4. IMPLEMENTATION ISSUES OF THE ADAPTIVE SIGN ALGORITHM.

The developed adaptive step-size algorithm will have a meaning only if it maintains the multiplication-free advantage of the sign algorithm. In this section we will show that the proposed algorithm in fact can be implemented in digital circuits with no real multiplication.

To see that, let us go back to the adaptation loop shown in Fig.3. Its output $f(i)$ is described by (9), namely,

$$f(i) = \alpha^{s_2(i)} f(i-1).$$

Let us set $\alpha = 2$ and $\Delta = 2$, where Δ is the step-size of the quantizer Q_2. Then from (8) and (9)

$$s_2(i) = \pm 1$$

and

$$f(i) = \begin{cases} 2f(i-1), & \text{if } s_2(i) = +1 \\ \frac{1}{2}f(i-1), & \text{if } s_2(i) = -1. \end{cases} \tag{13}$$

If we assume that $f(0)$ is a power-of-two number, then $f(i)$ is also power-of-two $\forall\ i$. Therefore, the binary representation of $f(i)$ always consist of one binary digit that is "1" and the rest are "0". Furthermore, this digit will simply shift one bit to the left or right depending on the value of $s_2(i)$. This process is illustrated in Fig. 5.

Since $f(i)$ is always a power-of-two number, the multiplication $s(i)f(i)u_i^*$ appears in Fig. 4 can be simply conducted by shifting u_i^* by the amount in $f(i)$ and then keeping or flipping the sign of the result depending on the value of $s(i)$.

Therefore, we can conclude that if $\alpha = 2$ and $\Delta = 2$, then the proposed adaptive sign algorithm can be implemented digitally simply by shift registers and with no real multiplication.

5. SIMULATION

In this section, we show the performance of the proposed adaptive sign algorithm using simulations on Matlab. The algorithm is used in the identification problem of a 4-tap FIR system with desired weight vector of $w_d = [5.1\ 3.2\ 2.5\ 1.5]^T$ starting from initial weights $w_0 = [50\ 50\ 50\ 50]$. The exponent constant α is set to 2 while the step-size μ is chosen arbitrarily as 0.1.

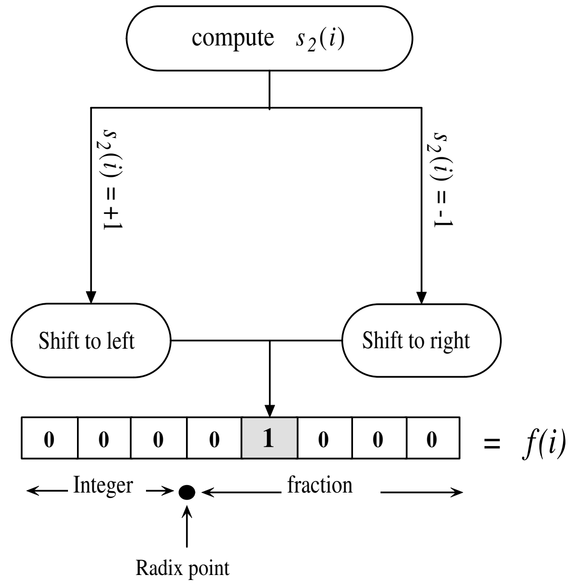

Fig. 5. *Implementation of the adaptation loop using a shift register.*

Fig.6 shows the learning curves of the LMS, sign algorithm (SLMS), and the proposed adaptive sign algorithm (ASLMS) over 100 runs. In this case the process noise is set to zero. The proposed algorithm shows superior performance compared to the sign algorithm. The performance of the proposed algorithm is comparable to the original LMS algorithm. In this case, the adaptation scheme almost completely eliminated the effect of the sign function on the recursion.

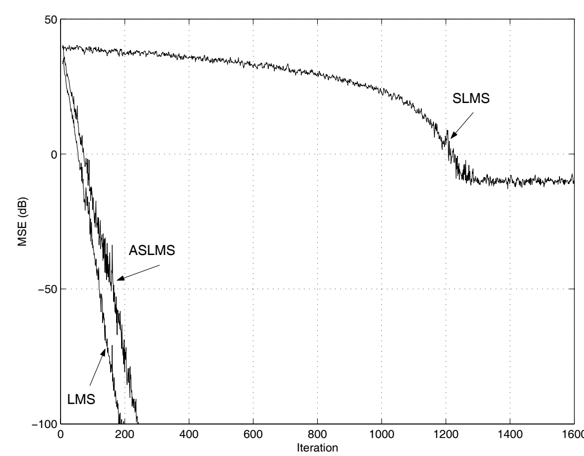

Fig. 6. *Learning curves of the proposed algorithm compared to sign algorithm (SLMS) and LMS algorithm with no process noise.*

The experiment is repeated but with process noise variance (σ_v^2) of -40dB. The idea here is to see if this noise will have an impact on the performance of the proposed algorithm. The resulting MSE curve is shown in Fig.7. Again, the proposed algorithm shows a performance close to the LMS algorithm, which is way better than that of the sign algorithm.

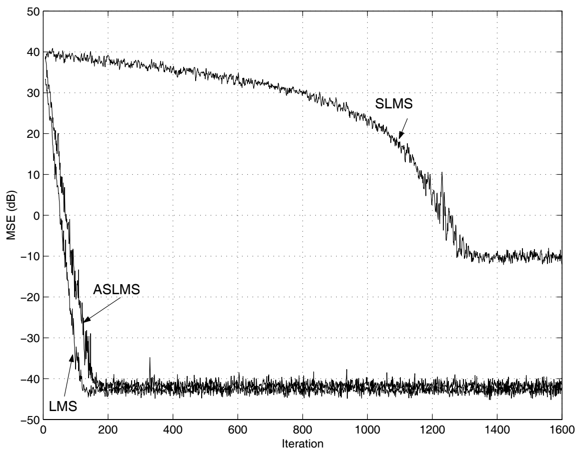

Fig. 7. *Learning curves of ASLMS, SLMS, and LMS algorithms with process noise.*

6. CONCLUSION

In this work, we proposed a novel structure for adaptive sign least mean squares (ASLMS). A powerful adaptation structure was used to adapt the step-size of the sign function inside the recursion of the sign algorithm. It was shown how the proposed algorithm can be implemented with no real multiplication. Simulations showed that the proposed algorithm has a superior performance compared to the sign algorithm. The adaptation loop almost completely eliminated the effect of the sign function resulting in a performance close to that of the original LMS algorithm.

7. REFERENCES

[1] A. H. Sayed, ”Fundamentals of Adaptive Filtering,” *Wiley, NY,* 2003.

[2] B. Farhang-Boroujeny, ”Adaptive Filters: theory and applications,” John Wiley and Sons Ltd, Baffins Lane, Chichester, UK, 2000.

[3] E. Eweda, ”Convergence analysis of the sign algorithm without the independence and gaussian assumptions,” IEEE Trans. Signal Processing, Vol. 48, No. 9, pp. 2535–44, Sept. 2000.

[4] E. Masry and F. Bullo, ”Convergence analysis of the sign algorithm for adaptive filtering,” IEEE Trans. Information Theory, Vol. 41, No. 2, pp. 489–95, March 1995.

[5] J. Treichler, C. Johnson, Jr., and M. Larimore, ”Theory and Design of Adaptive Filters,” Printice Hall, Inc., NJ, 2001.

[6] M. A. Aldajani and A. H. Sayed, “Stability and performance analysis of an adaptive sigma-delta modulator,” *IEEE Trans. on Circuits and Systems II: Analog and Digital Signal Processing,* vol. 48, no. 3, pp. 233-244, Mar. 2001.

[7] M. A. Aldajani and A. H. Sayed, “A stable adaptive structure for delta modulation with improved performance,” *Proc. ICASSP,* Salt Lake City, Utah, Vol. 4, pp. 2621–24, May 2001.

[8] M. A. Aldajani and A. H. Sayed, “SNR performance of an adaptive sigma delta modulator,” *Proc. IEEE International Symposium on Circuits and Systems,* Sydney, Australia, vol. 1, no. 5, pp. 392-395, May 2001.